REMARKS

Applicant thanks the examiner for their detailed review.

Status of Claims

Claims 1,2, 3, 7, 8, 12, 16, 19, 21, 23, and 34-36 have been amended. Claim 20 has been cacnelled.

Claim Rejections -35 USC § 102(b)

The Office Action States:

> 4b. **Claims 1-6, 10-15, 19, 20, and 34-35,** are rejected under 35 U.S.C. 102(e) as being anticipated by Grun (US pat. 6,629,166).

"[F]or anticipation under 35 U.S.C. 102, the reference must teach *every aspect* of the claimed invention …" MPEP 706.02 (emphasis added). "The identical invention must be shown *in as complete detail as contained in the ... claim*." *Richardson v., Suzuki Motor Co*., 868 F. 2d 1226, 1236, 9 USPQ2d 1913, 1920 (Fed. Cir. 1989) (emphasis added).

Applicant's amended claim 1 includes, "receiving a plurality of **partial data write transactions**, each of the plurality of partial data write transactions including **a write combinable attribute to indicate they are write combinable** partial data write transactions: **combining partial data** associated with the plurality of partial data write transactions in a buffer of an input/output (I/O) hub to **form write combined data**," (redacted claim 1 with emphasis). Grun fails to disclose at least the emphasis sections that applicant has bolded above, which are described in more detail below in reference to the three points of emphasis below.

First, Grun does not disclose any message or request that includes partial data write transactions, as in applicant's claim 1. To illustrate support in applicant's specification for this amendment, paragraph 0020, as an example, discloses combining partial writes into a cache line. Note that this support is purely exemplary, as their numerous other places in the specification that

disclose partial write transactions. In this embodiment, multiple partial writes are combined into a single write operation on a cache line basis to "achieve optimum burst performance." In contrast, Grun merely discloses collecting data for a message in a buffer and transferring control of that buffer between initiator's and I/O devices. However, nowhere does Grun disclose combining of multiple messages, such as partial data messages, into a single write/flush operation.

Second, Grun does not disclose write transactions include a write combinable attribute, as in applicant's claim 1. Again, an example of the support may be found in paragraph 0020 of applicant's specification, where a write-combinable (WC) attribute is associated with each page in memory…and if the writes have the WC attribute, the write transactions are sent out as write combinable. The Office Action points to col. 12 lines 10-12 of Grun, which discloses a format for a message primitive, i.e. a buffer ID field and a service connection field. Note from Grun's disclosure that the service connection ID field is to identify a specific initiator (See col. 11 lines 60-65) and the buffer ID is to identify/reference a buffer (See col 12 lines 12-15). As can be seen, Grun only discloses identification/reference to an initiator and a buffer, neither of which indicate a combinable nature of the message. Here, applicant allows for some messages not to be combined, i.e. the write attribute is not set or included as part of a write, or in contrast, as disclosed in claim 1, allows for a write to be separately identified as write combinable. Grun discloses no such ability, and merely discloses the ability to identify a source initiator and a buffer holding the message.

Third, Grun does not disclose combining partial data associated with the plurality of partial data write transactions in a buffer of an input/output (I/O) hub to form write combined data, as in applicant's claim 1. The Office action states that it Is known in the art that a message, request, command, or data is combined with an address. Applicant agrees with the office action. However, applicant's claim 1 is not discussing the combination of address with data, but rather the combination of partial data from a **plurality of write transactions**. Note that Grun's buffer

combines data with a service connect ID and a buffer ID for a message. However, Grun does not disclose anywhere combination of a plurality of partial data elements for a plurality of write transactions to form combined data, but rather only data for a single request message being combined with associated service and buffer identifiers.

Grun only describes the format for a single message and physical movement of data for "the" message (singular usage of message) are "left to a physical implementation of a target service interface (col. 12 lines 15-20). In stark contrast, applicant's claim 1 includes a plurality of write transactions. The data from the plurality of transactions are stored in a buffer to form "write combined data." As can be seen, write combined data does not include formulating a traditional packet (message, request, command, or data) as suggested in The Office Action, but rather the combining of data from a plurality of write transactions, which may include a plurality of packets.

In addition, Grun does not disclose identifying primitives or messages as write combinable, as described in applicant's claim 1, i.e. write transactions "are identified by the processor as write combinable." Note in paragraph 0020 of applicant's specification, an embodiment is described where "processor 16 does not have to actually combine the writes…the writes are sent out as write combinable." As a result, a processor may transmit the plurality of write transactions identifying them as write combinable and allow an I/O hub to perform the write combining. As discussed above, Grun only discloses sending a single message, selecting a pointer for control of a buffer, and once a complete message is received, passing the pointer back to an I/O device (col. 11 lines 36-39). No where does Grun discuss identifying transactions as write combinable or storing a plurality of messages to form combined data, as described in applicant's claim 1.

Applicant's claim 12 includes, "the first and the second write transactions to reference **partial data of a cache line** within the processor, wherein the first and second write transactions **include a write combinable attribute** to indicate the first and the second partial write transactions

as write combinable, **combine logic coupled to the receiving logic to combine the partial data** referenced by the first and second write transactions as write combined data," (claim 12 redacted with emphasis). Similar to the discussion above, Grun does not disclose applicant's combine logic to combine partial data from more than one message, i.e. a first write transaction and a second write transaction, but rather only assembling a single message, i.e. data with address/identifiers. Furthermore, as stated above, Grun discloses a service ID to reference an initiator and a buffer ID to reference a buffer, but does not disclose a write combinable attribute, as in applicant's claim 1. Additionally, nowhere does Grun disclose that multiple write reference partial data of a single cache line, but instead are only associated with an initiator ID to indicate a full initiator device.

Applicant's claim 34 has been amended to include, "page table logic to identify a page in a memory; wherein the page table logic is to associate a write combining attribute with the page in the memory to indicate that partial writes from the page are combinable." The Office Action group claim 34 with claim 1. However, Applicant fails to see where Grun discloses page table logic to identify a page in a memory, as in applicant's claim 1. Beyond identification of a page, applicant also fails to see where Grun discloses associating a write combining attribute with the page in memory or that a combining attribute is to indicate that partial writes from the page are combinable.

Claim Rejections -35 USC § 103(a)

The Office Actions states:

15. **Claims 7-9, 16-18, 31-33, and 36**, are rejected under 35 U.S.C. 103(a) as being unpatentable over Grun (US pat. 6,629,166) in view of Graham et al. (US pat. 6,223,641).

"The examiner bears the initial burden of factually supporting any prima facie conclusion of obviousness." MPEP § 2142. It is well established that *prima facie* obviousness is only established when three basic criteria are met. First, there must be some suggestion or motivation,

either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) must teach or suggest all the claim limitations. *In re Vaeck*, 947 F.2d 488 (Fed. Cir. 1991) (MPEP 2144).

Applicant's claim 31 includes, "determining whether a latency condition exists, the latency condition **including a delay in receiving a next combinable write transaction** from the processor **and an interface to the I/O device being in an idle state**," (claim 31 emphasis added). The Office Action cites col. 27 lines 9-15 of Grun for this element of claim 31. However, this section only describes purging all operations, i.e. not completing the operations by writing data but purging them, when a particular service connection is lost. After the purge the target channel adapter is returned to an idle condition. In contrast, applicant performs the combined write, i.e. flushes the data to the I/O device, in response to the latency condition, i.e. both a delay in the next combinable write and an idle of the interfaces to the I/O device. Note these are preconditions for the flush, i.e. a delay of the next write and the interface is idle coincide before the flush occurs in response to the latency condition. The Office Action also cites col. 6 lines 52-57, which describes a retry operation, where an operation is retried if determined that it is delayed. However, neither of the references alone or in combination disclose both a delayed **next write combinable** operation **and** an idle state of **the interface** to the I/O device occurring for a latency driven flush, as in applicant's claim 31.

The Office Action further states:

22. **Claims 21- 30**, are rejected under 35 U.S.C. 103(a) as being unpatentable over Grun (US pat. 6,629,166) in view of Kelly et al. (US pub. 2004/0019729).

Applicant's claim 21 includes, "a processor to associate a write combinable attribute with ~~identify~~ a plurality of write transactions to identify them as write combinable… the I/O hub having a write combining module to… combine ~~store~~ the partial data associated with the plurality of write transactions to form a write combined data set," (claim 21 redacted with emphasis). As stated above, Grun does not disclose associating a write combinable attribute with write transactions, but rather only combining a service connection ID to identify a specific initiator and a buffer ID to identify a buffer, neither of which convey a write combinable nature of an associated message. Furthermore, Grun's disclosure of buffers is to assemble a message, i.e. data for a single message with the buffer ID and service connection ID and/or address. However, Grun never discusses usage of partial data or combining data from multiple messages, as in applicant's claim 21.

Conclusion

Therefore, Applicants respectfully submit that claims 1, 12, 21, 31, and 34 are in condition for allowance, and furthermore, that dependent claims 2-11, 13-19, 22-30, 32-33, and 35-36 are also in condition for allowance for at least the same reasons stated above.

If there are any additional charges, please charge Deposit Account No. 50-0221. If a another telephone interview would in any way expedite the prosecution of the present application , the Examiner is invited to contact David P. McAbee at (503) 712-4988.

Respectfully submitted,
Intel Corporation

Dated: January 9, 2009

/David P. McAbee/Reg. No. 58,104
David P. McAbee
Reg. No. 58,104

Intel Corporation
M/S JF3-147
2111 NE 25th Avenue
Hillsboro, OR 97124
Tele – 503-712-4988
Fax – 503-264-1729